# 有限体上の超楕円曲線の位数計算を目的とするbaby step giant step algorithmの改良（暗号への応用を考慮した）

松尾 和人
（ 中央大学研究開発機構 ）



## 背景

$p$ : 奇素数
$\mathbb{F}_q$ : 有限体, $\mathrm{char}(\mathbb{F}_q) = p$, $\#\mathbb{F}_q = q$
$g$ : 正整数

$\mathbb{F}_q$ 上の genus $g$ の超楕円曲線 $C$

$$C : Y^2 = F(X),$$
$$F(X) = X^{2g+1} + f_{2g}X^{2g} + \cdots + f_0$$

$f_i \in \mathbb{F}_q, \mathrm{disc}\,(F) \neq 0$

$g = 2$とする

$\mathcal{J}_C$ : Cの Jacobi 多様体

$\mathcal{J}_C(\mathbb{F}_q)$ 上の離散対数問題:
$\mathcal{D}_1, \mathcal{D}_2 \in \mathcal{J}_C(\mathbb{F}_q) \to m \in \mathbb{Z}$ s.t. $\mathcal{D}_1 = m\mathcal{D}_2$

これを用いて離散対数ベースの暗号を構成したい



## $\mathcal{J}_C(\mathbb{F}_q)$ 上の離散対数問題に対する攻撃法

1. Square–root attack (＋Pohlig–Hellman)
2. Frey–Rück attack
3. Rück attack
4. Adleman–DeMarrais–Huang attack
5. Gaudry attack
6. Duursma attack
7. Weil descent attack



Square–root attack に対して安全であるには

$\#\mathcal{J}_C(\mathbb{F}_q) = cP$, $P$ : 160 bitより大きい素数

が必要

実装効率等を考慮すると $c$ は小さい方が良い
$(c = 1)$

$\Rightarrow$

超楕円曲線暗号を構成するために

$\#\mathcal{J}_C(\mathbb{F}_q) = P$, $P$ : 160 bitより大きい素数

なる $C$ が必要

## 素位数曲線の構成

**Input:** genus等の情報
**Output:** 素位数曲線$C$と$\#\mathcal{J}_C(\mathbb{F}_q)$
1: $C/\mathbb{F}_q$を選択
2: $\#\mathcal{J}_C(\mathbb{F}_q)$を計算
3: $\#\mathcal{J}_C(\mathbb{F}_q) \neq$ primeならばStep1へ

1. 特別な性質を持つ曲線を用いる方法
   - CM体法 (Frey, 高島, 中大)
   - Koblitz (Koblitz, 金山–長尾–内山)
2. ランダムな曲線を用いる方法
   - AGM (Harley–Mestre)
   - Kedlaya (Kedlaya, Gaudry)
   - Schoof–like
     (Pila, Kampkötter, Adleman–Huang)



## Gaudry–Harleyの Schoof–like algorithm

Gaudry, Harley,
*Counting points on hyperelliptic curves over finite fields*,
ANTS-IV
:Schoof–like algorithmの実装

$g = 2$の超楕円曲線について

- $\mathbb{F}_p$上127 bit位数
  ($p$: 63 bit)
- $\mathbb{F}_q$上128 bit位数
  ($p$: 16 bit, $q = p^4$)

を計算



Gaudry–Harley algorithm
**Input:** genus 2 HEC $C/\mathbb{F}_q$
**Output:** $\#\mathcal{J}_C(\mathbb{F}_q)$
1: $\#\mathcal{J}_C(\mathbb{F}_q) \bmod 2^e$ (Halving algorithm)
2: **for** 素数 $l = 3, 5, \ldots, l_{max}$ **do**
3: $\quad \chi_q(X) \bmod l$ (Schoof-like algorithm)
4: $\quad \chi_q(X) \bmod l \to \#\mathcal{J}_C(\mathbb{F}_q) \bmod l$
5: **end for**
6: $\chi_q(X) \bmod p$ (Cartier-Manin operator)
7: $\chi_q(X) \bmod p \to \#\mathcal{J}_C(\mathbb{F}_q) \bmod p$
8: $\#\mathcal{J}_C(\mathbb{F}_q) \bmod m,\ m = 2^e \cdot 3 \cdots l_{max} \cdot p$
(CRT)
9: $\#\mathcal{J}_C(\mathbb{F}_q) \bmod m \to \#\mathcal{J}_C(\mathbb{F}_q)$
(Square–root algorithm)

$\chi_q(X)$: $\mathcal{J}_C$の$q$乗Frobenius mapの特性多項式

Menezesが楕円曲線の位数計算に用いたalgorithmを超楕円曲線の位数計算に適用



## 127 bit位数の計算時間

| $l$ | $\#\mathcal{J}_C(\mathbb{F}_p) \bmod l$ | Time | CPU |
|---|---|---|---|
| $2^8$ | 176 | 12h | †‡ |
| 3 | 2 | 20m | † |
| 5 | 1 | 5m | † |
| 7 | 6 | 12h | † |
| 11 | 1 | 19h | † |
| 13 | 7 | 8d 13h | † |
| sqare-root algorithm | | 50d | ‡ |

†: Pentium 450 MHz (Magma)
‡: Alpha 500 MHz

暗号への応用を考慮したときの問題点

## 遅い

素位数曲線を発見するには数十回の位数計算が必要

## 本研究の内容

square–root algorithmの高速化



## $\#\mathcal{J}_C(\mathbb{F}_q) \bmod m \to \#\mathcal{J}_C(\mathbb{F}_q)$

$\mathcal{D} \in \mathcal{J}_C(\mathbb{F}_q)\backslash\{0\}$をランダムに選択し，

$$N\mathcal{D} = 0$$

を満足する$N \in \mathbb{Z}$をHasse–Weil range

$$L_o = \left\lceil (\sqrt{q}-1)^4 \right\rceil \le N \le H_o = \left\lfloor (\sqrt{q}+1)^4 \right\rfloor$$

の中で探す．

$$R = H_0 - L_0 = 8q^{3/2} + O(q)$$

Brute force: $O(q^{3/2})$

Baby step giant step: $O(q^{3/4})$

$\#\mathcal{J}_C(\mathbb{F}_q) \bmod m$を用いると高速化可能



$N_r \in \mathbb{Z}$ : given s.t.

$$\#\mathcal{J}_C(\mathbb{F}_q) = N_r + mN_m, 0 \le N_r < m$$

$\Rightarrow N_m$を

$$\lfloor L_o/m \rfloor \le N_m \le \lfloor H_o/m \rfloor$$

の中で決定すれば$\#\mathcal{J}_C(\mathbb{F}_q)$が求まる．

$$N_m = i + nj,\ n \in \mathbb{Z}, n \approx \sqrt{R/m}$$

$$0 \le i < n,$$
$$\left\lfloor \frac{L_o}{mn} \right\rfloor - 1 \le j \le \left\lfloor \frac{H_o}{mn} \right\rfloor$$

$\Rightarrow i, j$ともにrangeは$O(\sqrt{R/m})$

$$\#\mathcal{J}_C(\mathbb{F}_q)\mathcal{D} = (N_r + m(i+nj))\mathcal{D} = 0$$

$$(N_r + mi)\mathcal{D} = -mnj\mathcal{D}$$

計算量: $O(q^{3/4}/\sqrt{m})$

※ Gaudry–Harleyが用いたalgorithmも同一計算量



## 高速化

$$\chi_q(X) = X^4 - s_1X^3 + s_2X^2 - s_1qX + q^2 \in \mathbb{Z}[X],$$

$$|s_1| \le 4\sqrt{q},$$
$$|s_2| \le 6q$$

Halving algorithm : 素位数曲線に対し有効でない

Schoof–like algorithm : $s_i \bmod l$を計算可能

Cartier–Manin operator : $s_i \bmod p$を計算可能

$s_i \bmod m$を利用すれば
baby step giant step algorithmを
高速化できるのではないか？

$s_1 \times s_2$の面積$\approx 96q^{3/2}$

**Lemma 1.** $s_1$は

$$s_{1l} = -\lfloor 4\sqrt{q} \rfloor \le s_1 \le s_{1u} = \lfloor 4\sqrt{q} \rfloor$$

に値をとる．また，$s_2$は

$$s_{2l} = \lceil 2\sqrt{q}|s_1| - 2q \rceil \le s_2 \le s_{2u} = \left\lfloor \frac{1}{4}s_1^2 + 2q \right\rfloor$$

に値をとる．

$s_{2u}$ : N. D. Elkies,
*Elliptic and modular curves over finite fields and related computational issues*, Computational perspectives on number theory (D. A. Buell and J. T. Teitlbaum, eds.), AMS, 1995, pp. 21–76.
$s_{2l}$ : 百瀬，私信

$$\int \frac{1}{4}s_1^2 + 2q - (2\sqrt{q}|s_1| - 2q)\mathrm{d}s_1 = s_1(\frac{1}{12}s_1^2 - \sqrt{q}|s_1| + 4q)$$

$s_1 \times s_2$の面積$\approx \frac{32}{3}q^{3/2}$



$$0 \le t_2 < n$$
$$\left\lfloor \frac{s_{2l}}{mn} \right\rfloor - 1 \le t_3 \le \left\lfloor \frac{s_{2u}}{mn} \right\rfloor$$

$$\begin{aligned}\#\mathcal{J}_C(\mathbb{F}_q) &= \chi_q(1) \\ &= q^2 + 1 - s_1(q+1) + s_2 \\ &= q^2 + 1 - s_1'(q+1) + s_2' \\ &\quad - m(q+1)t_1 + mt_2 + mnt_3\end{aligned}$$

$$\begin{aligned}(q^2+1-s_1'(q+1)+s_2'-m(q+1)t_1+mnt_3)\mathcal{D} \\ = -mt_2\mathcal{D}\end{aligned}$$

今後，右辺の計算をbaby step，左辺の計算をgiant stepと呼ぶ．

計算量: $O(q^{3/4}/m)$

Gaudry–Harleyが用いた方法より$O(\sqrt{m})$倍高速



## $s_i \bmod m \to \#\mathcal{J}_C(\mathbb{F}_q)$

$s_i' \in \mathbb{Z}$ : given s.t.

$$0 \le s_i' < m,$$
$$s_1 = s_1' + mt_1, t_1 \in \mathbb{Z},$$
$$s_2 = s_2' + mt_2', t_2' \in \mathbb{Z}$$

$$\left\lfloor \frac{s_{1l}}{m} \right\rfloor \le t_1 \le \left\lfloor \frac{s_{1u}}{m} \right\rfloor$$
$$\left\lfloor \frac{s_{2l}}{m} \right\rfloor \le t_2' \le \left\lfloor \frac{s_{2u}}{m} \right\rfloor$$

$n \in \mathbb{Z}$ :

$$n \approx \frac{4\sqrt{6}q^{3/4}}{3m}$$
$$t_2' = t_2 + nt_3,\ t_2, t_3 \in \mathbb{Z}$$



## 実装

Input: genus 2 HEC $C/\mathbb{F}_q$
Output: $\#\mathcal{J}_C(\mathbb{F}_q)$

1. $s_i \bmod 2$
2. $s_i \bmod p$ (Cartier-Manin operator)
3. $s_i \bmod m,\ m = 2p$ (CRT)
4. $s_i \bmod m \to \#\mathcal{J}_C(\mathbb{F}_q)$
   (Proposed baby step giant step algorithm)

これを素位数曲線が見付かるまで繰り返した。

実際には,

**Lemma 2.**

$2 \nmid \#\mathcal{J}_C(\mathbb{F}_q) \Leftrightarrow F : \textit{irreducible}/\mathbb{F}_q \Leftrightarrow 2 \nmid s_i$

より, $F$が既約な$C$を入力し

$s_i \equiv 1 \bmod 2$

とした.

また,
Cartier–Manin operator,
Baby step giant step algorithm
の計算にも高速化手法を用た.

加算は,
M, Chao, Tsujii, *Fast genus two hyperelliptic curve cryptosystems*, Technical Report ISEC2001-31, IEICE Japan, 2001.
に記載のHarley algorithm

有限体の元と多項式の演算にNTLを使用した.



**Example 1.**

$p = 1342181$,
$\mathbb{F}_q = \mathbb{F}_p(\alpha)$,
$\alpha^3 + 1073470\alpha^2 + 34509\alpha + 1223366 = 0$

$C_1/\mathbb{F}_q : Y^2 = F_1(X)$,
$F_1 = X^5 + (567033\alpha^2 + 322876\alpha + 957805)X^4$
$+ (1123698\alpha^2 + 933051\alpha + 141410)X^3$
$+ (393269\alpha^2 + 233572\alpha + 708577)X^2$
$+ (692270\alpha^2 + 350968\alpha + 788883)X$
$+ 968896\alpha^2 + 895453\alpha + 589750$

$\#\mathcal{J}_{C_1}(\mathbb{F}_q) =$
584610376401469447932232931574028593 1

: 123 bit prime number

| | Time |
|---|---|
| Cartier–Manin operator | 7m |
| Baby step (26 bit) | 1h 10m |
| Sort | 1m |
| Giant step | 1h 59m |
| Total | 3h 17m |

Pentium III/866MHz, 1G RAM



**Example 2.**

$p = 5491813$,
$\mathbb{F}_q = \mathbb{F}_p(\alpha)$,
$\alpha^3 + 4519302\alpha^2 + 3749080\alpha + 607603 = 0$

$C_2/\mathbb{F}_q : Y^2 = F_2(X)$,
$F_2 = X^5 + (2817153\alpha^2 + 3200658\alpha + 1440424)X^4$
$+ (3310325\alpha^2 + 481396\alpha + 1822351)X^3$
$+ (108275\alpha^2 + 120315\alpha + 469800)X^2$
$+ (2168383\alpha^2 + 1244383\alpha + 5010679)X$
$+ 4682337\alpha^2 + 53865\alpha + 2540378$

$\#\mathcal{J}_{C_2}(\mathbb{F}_q) =$
27434335457581234045473311611818187339271

: 135 bit prime number

| | Time |
|---|---|
| Cartier–Manin operator | 42m |
| Baby step (28 bit) | 5h 30m |
| Sort | 20m |
| Giant step | 9h 17m |
| Total | 15h 49m |

Alpha 21264/667MHz, 4G RAM



## まとめ

- 超楕円曲線の位数計算を目的とし
  baby step giant step algorithm
  の改良を行った.
- 提案algorithmを用いて135 bitの素位数曲線を構成できた.
- Memory使用量削減は今後の課題
- Gaudry–Harleyの方法と併せて用いることで
  160 bit位数の計算は可能と考えられる.
- しかし, 160 bit素位数曲線の構成を行うためには
  Schoof-like algorithmの改良が必要

## Appendix

Magma V.2.8:
Gaudry–Harley's Schoof–like algorithmの
Gaudry自身が書いたcodeを含む.

**Example 3.**

$p = 2^{20} - 5$,
$\mathbb{F}_q = \mathbb{F}_p(\alpha)$,
$\alpha^4 + 278680\alpha^3 + 445675\alpha^2 + 218811\alpha + 653340 = 0$

$C_3/\mathbb{F}_q : Y^2 = F_3(X)$,
$F_3 = X^5$
$+ (508797\alpha^3 + 672555\alpha^2 + 940125\alpha + 153314)X^3$
$+ (330843\alpha^3 + 367275\alpha^2 + 910087\alpha + 1002854)X^2$
$+ (488395\alpha^3 + 873290\alpha^2 + 734350\alpha + 7072)X$
$+ 180553\alpha^3 + 25142\alpha^2 + 806296\alpha + 724502$

$\#\mathcal{J}_{C_3}(\mathbb{F}_q)$ =1461445886397612447866396786769393107114349704111
=37 × 79 × 6055499440163
× 825665152652002064231054502874 39

: 160 bit



| | Time |
|---|---|
| $l = 3$ | 27s |
| $l = 5$ | 14m 46s |
| $l = 7$ | 3h 10m 37s |
| $l = 11$ | 20d 20h 23m 38s |
| Cartier–Manin operator | 10m 42s |
| Baby step (30 bit) | 1d 23h 22m 22s |
| Sort | 2h 5m 15s |
| Giant step | 2d 23h 3m 34s |
| Total | 26d 19h 31m 21s |

Schoof–like algorithm: Pentium III/866MHz, 1G RAM
The others: Itanium/800MHz, 12G RAM